App. Ser. No.: 10/017,959 Attv. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

IN THE CLAIMS:

Please amend the claims as follows:

1. (Previously Presented) A computer-implemented method of debugging code containing a user-specified breakpoint located within a predetermined region of the code, the method comprising:

setting a machine recognizable entry point in the code;

setting a machine recognizable exit point in the code, wherein the entry point and the exit point define an entry and an exit, respectively, of the region;

executing the code;

entering the region by encountering the machine recognizable entry point during the execution;

determining whether the execution of the code exits the region of the code without firing the user-specified breakpoint; exiting the region being determined by encountering the machine recognizable exit point during the execution; and

if so, halting the execution of the code.

- 2. (Previously Presented) The method of claim 1, wherein the user-specified breakpoint is a conditional breakpoint having an associated condition and wherein execution exits the region without firing the user-specified breakpoint because the associated condition is not satisfied.
- (Previously Presented) The method of claim 1, further comprising: encountering the user-specified breakpoint; suspending the execution of the code at the user-specified breakpoint; and

in response to a user-specified run-to command received while execution of the code is suspended, executing the code until reaching the machine recognizable exit point of the region.

App. Ser. No.: 10/017,959 Atty. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

- 4. (Previously Presented) The method of claim 1, wherein the user-specified breakpoint is a non-conditional breakpoint and wherein execution exits the region without firing the user-specified breakpoint because the user-specified breakpoint is not
- encountered.
- 5. (Previously Presented) The method of claim 1, wherein:

setting the machine recognizable entry point in the code comprises setting an internal safety net entry breakpoint in the code; and

setting a machine recognizable exit point in the code comprises setting a safety net exit breakpoint in the code.

- 6. (Original) The method of claim 5, wherein setting the internal safety net entry breakpoint and setting the safety net exit breakpoint are performed automatically in response to a user selection of the region.
- 7. (Previously Presented) The method of claim 5, wherein setting the safety net exit breakpoint is performed automatically in response to encountering the internal safety net entry breakpoint.
- 8. (Previously Presented) The method of claim 5, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.
- 9. (Previously Presented). The method of claim 1, wherein halting comprises encountering a safety net breakpoint located in the code relative to the machine recognizable exit point.
- 10. (Previously Presented) The method of claim 1, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.

Page 3

App. Ser. No.: 10/017,959 Atty. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

11. (Previously Presented) A computer readable medium containing a debug program which, when executed, performs an operation for debugging code containing a user-specified breakpoint located within a predetermined region of the code, the operation comprising:

defining the region of the code by a machine recognizable entry point and a machine recognizable exit point;

executing the code;

entering the region during the execution;

determining whether the execution of the code exits the region of the code without firing the user-specified breakpoint; exiting the region being determined by encountering the machine recognizable exit point during the execution; and

if so, halting the execution of the code.

- 12. (Previously Presented) The computer readable medium of claim 11, wherein the user-specified breakpoint is a conditional breakpoint having an associated condition and wherein execution exits the region without firing the user-specified breakpoint because the associated condition is not satisfied.
- 13. (Previously Presented) The computer readable medium of claim 11, wherein the user-specified breakpoint is a non-conditional breakpoint and wherein execution exits the region without firing the user-specified breakpoint because the user-specified breakpoint is not encountered.
- 14. (Previously Presented) The computer readable medium of claim 11, wherein defining the region comprises:

setting an internal safety net entry breakpoint in the code relative to the machine recognizable entry point of the region; and

setting a safety net exit breakpoint in the code relative to the machine recognizable exit point of the region.

Page 4

App. Ser. No.: 10/017,959 Atty. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

- 15. (Original) The computer readable medium of claim 14, wherein setting the internal safety net entry breakpoint and setting the safety net exit breakpoint are performed automatically in response to a user selection of the region.
- 16. (Previously Presented) The computer readable medium of claim 14, wherein setting the safety net exit breakpoint is performed automatically in response to encountering the internal safety net entry breakpoint.
- 17. (Previously Presented) The computer readable medium of claim 14, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.
- 18. (Previously Presented) The computer readable medium of claim 11, wherein halting comprises encountering a safety net breakpoint located in the code relative to the machine recognizable exit point.
- 19. (Previously Presented) The computer readable medium of claim 11, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.
- 20. (Previously Presented) A computer system, comprising:
- a memory containing at least a debug program and code containing a userspecified breakpoint located within a predetermined region of the code; and
- a processor which, when executing content of the memory, is configured to perform an operation comprising:

defining the region of the code by a machine recognizable entry point and a machine recognizable exit point;

executing the code;

App. Ser. No.: 10/017,959 Atty. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

determining whether the execution of the code exits the region of the code without firing the user-specified breakpoint; and

if so, halting the execution of the code upon reaching the machine recognizable exit point of the region.

- 21. (Previously Presented) The computer system of claim 20, wherein the user-specified breakpoint is a conditional breakpoint having an associated condition and wherein execution exits the region without firing the user-specified breakpoint because the associated condition is not satisfied.
- 22. (Previously Presented) The computer system of claim 20, wherein the user-specified breakpoint is a non-conditional breakpoint and wherein execution exits the region without firing the user-specified breakpoint because the user-specified breakpoint is not encountered.
- 23. (Previously Presented) The computer system of claim 20, wherein the operation further comprises, prior to determining whether the execution exits the code without firing the user-defined breakpoint:

setting an internal safety net entry breakpoint in the code relative to an the machine recognizable entry point of the region; and

setting a safety net exit breakpoint in the code relative to the machine recognizable exit point of the region.

- 24. (Original) The computer system of claim 23, wherein setting the internal safety net entry breakpoint and setting the safety net exit breakpoint are performed automatically in response to a user selection of the region.
- 25. (Previously Presented) The computer system of claim 23, wherein setting the safety net exit breakpoint is performed automatically in response to encountering the internal safety net entry breakpoint.

Page 6

App. Ser. No.: 10/017,959 Atty. Dkt. No. ROC920010308US1

PS Ref. No.: IBMK10308

- 26. (Previously Presented) The computer system of claim 23, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.
- 27. (Previously Presented) The computer system of claim 20, wherein halting comprises encountering a safety net breakpoint located in the code relative to the machine recognizable exit point.
- 28. (Previously Presented) The computer system of claim 20, wherein the machine recognizable entry point and the machine recognizable exit point are determined by a compiler.